STATS 509 Homework 4   Xiaofeng Nie

1. Solution:

(a)

$$Corr(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}$$

$$E(Y) = P(Z=1)E(XZ|Z=1) + P(Z=-1)E(XZ|Z=-1)$$
$$= \frac{1}{2}E(X) + \frac{1}{2}E(-X) = 0$$

$$Cov(X,Y) = E(XY) - E(X)E(Y) = E(X^2Z)$$
$$= P(Z=1)E(X^2|Z=1) + P(Z=-1)E(X^2|Z=-1)$$
$$= \frac{1}{2}E(X^2) + \frac{1}{2}E(-X^2) = 0$$

$$Var(Y) = E(Y^2) - (EY)^2 = E(X^2Z^2) - 0$$
$$= P(Z=1)E(X^2Z^2|Z=1) + P(Z=-1)E(X^2Z^2|Z=-1)$$
$$= E(X^2) = Var(X) = 1$$

Because $Var(X)$ and $Var(Y) \neq 0$, we know $Corr(X,Y) = 0$.

Because the correlation between X and Y is 0, X and Y are uncorrelated.

(b)

When $g(X) = X^2, h(X) = X^2$,

$$E\big(g(X)h(Y)\big) = E(X^2X^2Z^2) = E(X^4) = 3$$

$$E\big(g(X)\big)E\big(h(Y)\big) = E(X^2)E(X^2) = 1$$

Because $E\big(g(X)h(Y)\big) \neq E\big(g(X)\big)E\big(h(Y)\big)$, we know that X and Y are not independent.

(c)

$$F_Y(y) = P(XZ \le y) = \frac{1}{2}P(X \le y) + \frac{1}{2}P(-X \le y) = F_X(y)$$

Since $F_X(X)$ is uniform distributed in [0,1], $F_Y(Y)$ is also uniform distributed in [0,1].

Thus, $Var(F_X(X)) = Var\big(F_Y(Y)\big) = \frac{1}{12}$, $E(F_X(X)) = E\big(F_Y(Y)\big) = \frac{1}{2}$

$$\rho_s = \rho\big(F_X(X), F_Y(Y)\big) = \frac{cov\big(F_X(X), F_Y(Y)\big)}{\sqrt{Var\big(F_X(X)\big)Var\big(F_Y(Y)\big)}}$$

$$= \frac{E[F_X(X), F_Y(Y)] - E[F_X(X)]E[F_Y(Y)]}{\frac{1}{12}}$$
$$= 12E[F_X(X), F_Y(Y)] - 3$$
$$= 12 * \frac{1}{2}E[F_X(X), F_Y(X)] + 12 * \frac{1}{2}E[F_X(X), F_Y(-X)] - 3$$
$$= 6E[F_X(X)^2] + 6E\big[F_X(X)\big(1 - F_X(X)\big)\big] - 3$$
$$= 6E[F_X(X)] - 3 = 0$$

2. Solution:

For multivariate t-distribution, when $\nu > 2$:

$$Cov\big(Y_i, Y_j\big) = \frac{\nu}{\nu - 2}\Lambda_{ij}$$

$$Corr\big(Y_i, Y_j\big) = \frac{Cov\big(Y_i, Y_j\big)}{\sqrt{Var(Y_i)Var\big(Y_j\big)}} = \frac{\frac{\nu}{\nu-2}\Lambda_{ij}}{\sqrt{\frac{\nu}{\nu-2}\Lambda_{ii}\frac{\nu}{\nu-2}\Lambda_{jj}}}$$

$$= \frac{\Lambda_{ij}}{\sqrt{\Lambda_{ii}\Lambda_{jj}}} \quad \dots \dots \dots \dots \dots \dots \dots (1)$$

For multivariate normal distribution:

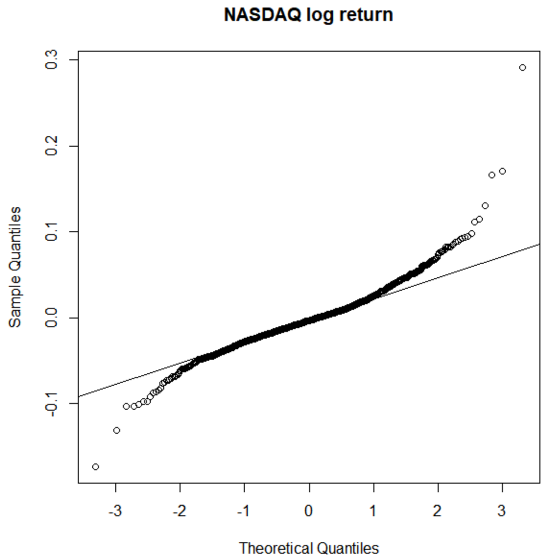
$$Cov(X_i, X_j) = \Lambda_{ij}$$

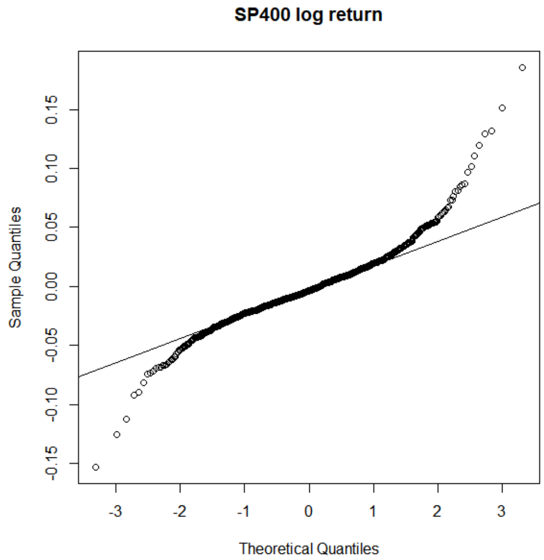
$$Corr(Y_i, Y_j) = \frac{\Lambda_{ij}}{\sqrt{\Lambda_{ii}\Lambda_{jj}}} \quad \dots \dots \dots \dots \dots \dots \dots (2)$$

Because (1) = (2), we know that ***X*** and ***Y*** have the same correlation matrices.

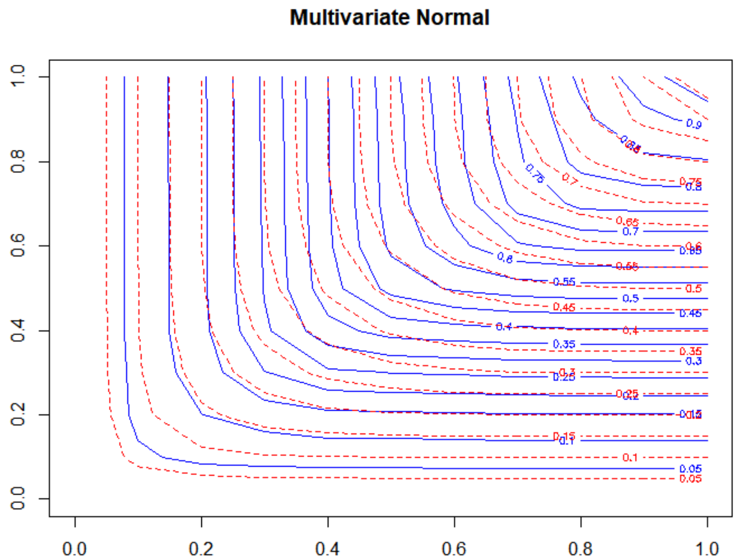
3. Solution:

(a) Univariate Q-Q plot shows that both of these two log returns are not normal distributed. There are heavy tails for both of them.



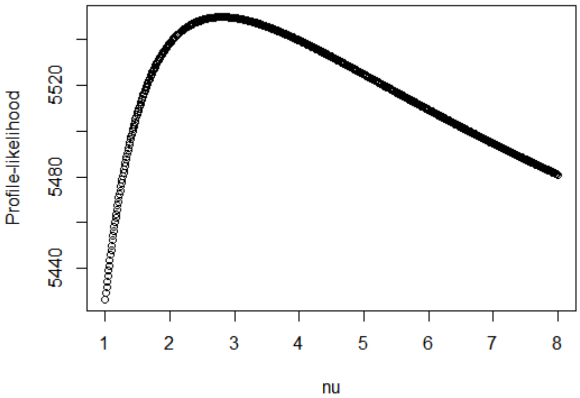


Lines showed in the plot below represent empirical bivariate-normal cumulative distribution function, while dots represent theoretical bivariate-normal cumulative distribution function. These lines show that the multivariate normal distribution is quite good for NASDAQ log return and SP400 log return, though univariate normal distribution is not good for them.

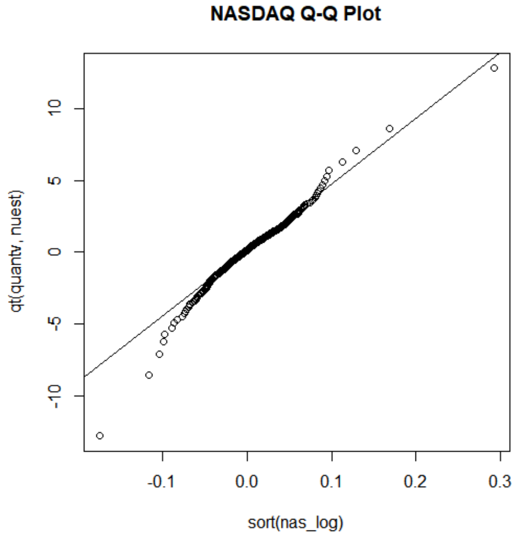


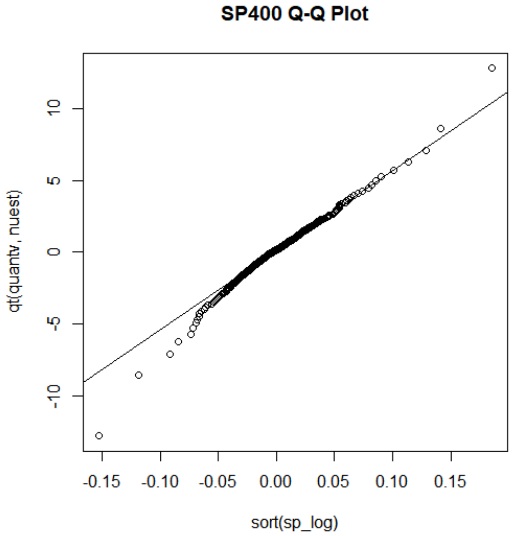
(b) In order to maximize pseudo likelihood, we should choose confidence interval of degree of freedom is [2.43,3.25]. And the best freedom degree for marginal distributions and multivariate-t distribution is 2.8

```
> c(lower_bound,upper_bound)   > nuest
[1] 2.44 3.25                  [1] 2.8
```

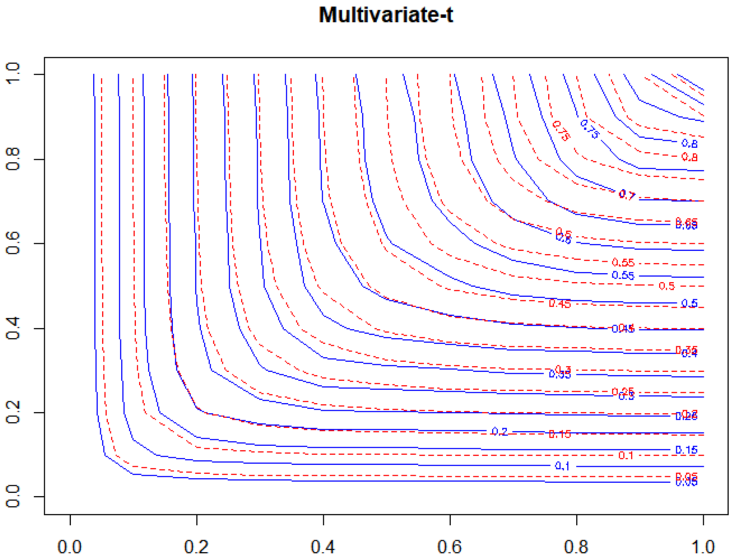


Univariate Q-Q plot shows that both of these two log returns can be t-distributed.





Lines showed in the plot below represent empirical bivariate-t cumulative distribution function, while dots represent theoretical bivariate-t cumulative distribution function. These lines show that the multivariate-t distribution is quite good for NASDAQ log return and SP400 log return.



(c) Considering plots above, we can find that multivariate-t distribution is obviously better. Not only because it's CDF plot is close to theoretical plot, but also marginal plots are better.

(d) In multivariate normal model, when weight is 0.5, VaR is 920424.

```
> VaR
   0.1%
920424
```

In multivariate-t distribution, when weight is 0.5, VaR is 1802270.

```
> VaR
   0.1%
1802270
```

(e) Using multivariate normal distribution:

```
> w_exp;w_var;wmax
[1] 1
[1] 0
[1] 0
```

-To maximize expected return, w=1, i.e. the portfolio is all of NASDAQ.
-To minimize volatility, w=0, i.e. the portfolio is all of SP400.
-To minimize VaR, when q=0.002, w=0, i.e. the portfolio is all of SP400.

Using multivariate-t distribution:

```
> w_exp;w_var;wmax
[1] 1
[1] 0
[1] 0.18
```

-To maximize expected return, w=1, i.e. the portfolio is all of NASDAQ
-To minimize volatility, w=0, i.e. the portfolio is all of SP400.
-To minimize VaR, when q=0.002, w=0.18, i.e. the portfolio consists of 18% NASDAQ and 82% million SP400.

Appendix:

R code for problem 1

```
library(mvtnorm)
library(copula)
library(MASS)
library(fCopulae)
library(fGarch)
library(mnormt)

nasdaq = read.csv("O:\\18WIN\\STATS 509\\HW4\\Nasdaq_wklydata_92-
12.csv",header = T)
sp400 = read.csv("O:\\18WIN\\STATS 509\\HW4\\SP400Mid_wkly_92-
12.csv",header = T)
nas_adjclose = nasdaq$Adj.Close
sp_adjclose = sp400$Adj.Close
nas_log = diff(log(nas_adjclose))
sp_log = diff(log(sp_adjclose))

#a) multivariate normal distribution
myxlim = c(-0.2,0.2)
myylim = c(-0.2,0.2)
mean_nas = mean(nas_log)
sd_nas = sd(nas_log)
mean_sp = mean(sp_log)
sd_sp = sd(sp_log)
mu = c(mean(nas_log),mean(sp_log))
mu
sigma = var(cbind(nas_log,sp_log))
sigma

n = length(nas_log)
nsim = rmvnorm(n,mu,sigma)
par(mfrow=c(1,2))
plot(nas_log,sp_log,xlim=myxlim,ylim=myylim,xlab="Nasdaq",ylab="SP400")
plot(nsim,xlim=myxlim,ylim=myylim,xlab="X-simulation",ylab="Y-
Simulation")
qqnorm(nas_log,main="NASDAQ log return")
qqline(nas_log)
```

```
qqnorm(sp_log,main="SP400 log return")
qqline(sp_log)
#empirical
cdf_nas = pnorm(nas_log,mean_nas,sd_nas)
cdf_sp = pnorm(sp_log,mean_sp,sd_sp)
dem = pempiricalCopula(cdf_nas,cdf_sp)
contour(dem$x,dem$y,dem$z,main="Multivariate Normal",col='blue',lty=1,lwd=1,nlevel=20)
#theoretical
theo_norm = mvrnorm(1e6,mu,sigma)
theo_1 = theo_norm[,1]
theo_2 = theo_norm[,2]
cdf_theo_1 = pnorm(theo_1,mean(theo_1),sd(theo_1))
cdf_theo_2 = pnorm(theo_2,mean(theo_2),sd(theo_2))
dem_theo = pempiricalCopula(cdf_theo_1,cdf_theo_2)
contour(dem_theo$x,dem_theo$y,dem_theo$z,main="Multivariate Normal",col='red',lty=2,lwd=1,add=TRUE,nlevel=20)

#b) multivariate t distribution
combination = cbind(nas_log, sp_log)
df = seq(1, 8, 0.01)
n = length(df)
loglik_max = rep(0, n)
for(i in 1:n){
  fit = cov.trob(combination, nu = df[i])
  mu = as.vector(fit$center)
  sigma = matrix(fit$cov, nrow = 2)
  loglik_max[i] = sum(log(dmt(combination, mean=fit$center, S=fit$cov, df=df[i])))
}
plot(df, loglik_max, xlab='nu', ylab='Profile-likelihood')
##df
nuest = df[which.max(loglik_max)]
nuest
##CI of df
position = which((loglik_max[which.max(loglik_max)]-loglik_max) <= 0.5*qchisq(0.95, 1))
lower_bound = df[position[1]]
upper_bound = df[position[length(position)]]
c(lower_bound,upper_bound)

par(mfrow=c(1,2))
quantv = (1/n)*seq(.5,n-.5,1)
qqplot(sort(nas_log),qt(quantv,nuest),main="NASDAQ Q-Q Plot")
abline(lm(qt(c(.25,.75),nuest)~quantile(nas_log,c(.25,.75))))
qqplot(sort(sp_log),qt(quantv,nuest),main="SP400 Q-Q Plot")
abline(lm(qt(c(.25,.75),nuest)~quantile(sp_log,c(.25,.75))))
#empirical
cdf_nas_t = pstd(nas_log,mean_nas,sd_nas,nuest)
cdf_sp_t = pstd(sp_log,mean_sp,sd_sp,nuest)
dem_t = pempiricalCopula(cdf_nas_t,cdf_sp_t)
contour(dem_t$x,dem_t$y,dem_t$z,main="Multivariate-t",col='blue',lty=1,lwd=1,nlevel=20)
#theoretical
mu = c(mean(nas_log),mean(sp_log))
sigma = var(cbind(nas_log,sp_log))
lambda = sigma*(nuest-2)/nuest
theo_t = rmt(1e6,mu,lambda,nuest)
theo_1 = theo_t[,1]
theo_2 = theo_t[,2]
cdf_theo_1 = pstd(theo_1,mean(theo_1),sd(theo_1),nuest)
cdf_theo_2 = pstd(theo_2,mean(theo_2),sd(theo_2),nuest)
dem_theo = pempiricalCopula(cdf_theo_1,cdf_theo_2)
```

```r
contour(dem_theo$x,dem_theo$y,dem_theo$z,col='red',lty=2,lwd=1,add=TRUE,
nlevel=20)

#d)-normal
mu = c(mean(nas_log),mean(sp_log))
sigma = var(cbind(nas_log,sp_log))
theo_norm = mvrnorm(1e6,mu,sigma)
datat = 0.5*theo_norm[,1]+0.5*theo_norm[,2]
VaR = -quantile(datat,0.001)*1e7
VaR
#d)-t
theo_t = rmt(1e6,mu,lambda,nuest)
datat = 0.5*theo_t[,1]+0.5*theo_t[,2]
VaR = -quantile(datat,0.001)*1e7
VaR

#e)-t
set.seed(2015)
w = seq(0,1,0.01)
n = length(w)
VaRv=rep(0,n)
exp_return = rep(0,n)
var=rep(0,n)
data_sim = rmt(1e4,mu,lambda,nuest)
for(i in 1:n){
  datat = w[i]*data_sim[,1]+(1-w[i])*data_sim[,2]
  VaRv[i] = -quantile(datat,0.002)
  exp_return[i] = mean(datat)
  var[i] = sd(nas_log)^2*w[i]^2+sd(sp_log)^2*(1-
w[i])^2+2*sd(nas_log)*sd(sp_log)*cor(nas_log,sp_log)*w[i]*(1-w[i])
}
w_exp = w[which.max(exp_return)]
exp_max = exp_return[which.max(exp_return)]*1e7
w_var = w[which.min(var)]
var_min = var[which.min(var)]
wmax = w[which.min(VaRv)]
VaR = VaRv[which.min(VaRv)]*1e7
w_exp;w_var;wmax

#e)-norm
set.seed(2015)
w = seq(0,1,0.01)
n = length(w)
VaRv=rep(0,n)
exp_return = rep(0,n)
var=rep(0,n)
data_sim = mvrnorm(1e4,mu,sigma)
for(i in 1:n){
  datat = w[i]*data_sim[,1]+(1-w[i])*data_sim[,2]
  VaRv[i] = -quantile(datat,0.002)
  exp_return[i] = mean(datat)
  var[i] = sd(nas_log)^2*w[i]^2+sd(sp_log)^2*(1-
w[i])^2+2*sd(nas_log)*sd(sp_log)*cor(nas_log,sp_log)*w[i]*(1-w[i])
}
w_exp = w[which.max(exp_return)]
exp_max = exp_return[which.max(exp_return)]*1e7
w_var = w[which.min(var)]
var_min = var[which.min(var)]
wmax = w[which.min(VaRv)]
VaR = VaRv[which.min(VaRv)]*1e7
w_exp;w_var;wmax
```